1 Background studies for “2γ in PR” case

2 Initial assumptions

Shortly, they are all the assumptions included in the version v3_0 of FastMC. More noticeable changes include a new photon veto (PV) inefficiency curve, energy dependent catcher inefficiency, other assumption a about charge veto.
See `http://www.phy.bnl.gov/~djaffe/KOPIO/fastmc/latest.version.html` for details.

3 Approach

The previous results were that with the Andrei's contour cut we had $\sim$ 125 events of kp2 background and $\sim$ 160 events of total background at $\sim$ 110 events of signal, but the cut was designed at slightly different assumptions about the detector performance. On the other hand a Neural Network (NN) cut was designed with up to date assumptions but trained only with kp2 background. Correspondingly NN cut did better with kp2 ($\sim$ 70 events of kp2 at the same $\sim$ 110 events of signal) but enormous amount of kcp3 background and significant amount of ke3g passed this cut.

This issues were addressed in the following studies.

3.1 The contour cut reoptimization

This was done in the following way. Events passed setup cuts below:

- fiducial ones
  - on Z_K
  - $P_K > 400$ GeV
- David's ones targeted odd and even high weight events
- $|M_{\gamma\gamma} - M_{\pi^0}| < 20$ MeV

were used separately for each decay mode to fill the contour variable bi-plots. Then tree major background plots (kp2, ke3g, kcp3) were normalized (to $N_K = 0.1271 \times 10^{16}$ after the spoiler) and then summed. Special care was taken to put errors to each bin approprietly. The signal plot was also normalized and the divided by the summed background. The obtained S/N distribution was then smoothed with the HBOOK multi-quadric smoothing algorithm. The result is shown in Fig. 1. The result of smoothing can also be written in the form of a Fortran code containing the corresponding function (let call it `hquadf`).

Obviously we should follow the contours of the plot to design the best cut which is the same as requiring `hquadf(x,y)`$> f_{thr}$.

The contour at the level of $f_{thr} = 0.3$ shown as a red line in the figure seems to resemble quite closely the Andrei's original contour cut. But now with the function it's easy to construct a set of cuts with different signal acceptance and to have an entire curve of amount of background vs amount of signal. Such a curve is shown in Fig. 2.

It seems to show that we can not do much better than what was already demonstrated with the original contour cut (although now there is the entire curve).

3.2 NN cut

Here we have to additionally suppress kcp3 and ke3g backgrounds. Since the contour cut seem to be doing well on them the idea was to use lower part of the contour as a fixed setup cut to suppress kcp3 and somewhat of ke3g and then retrain NN to suppress kp2. The cut off region can be seen in distributions in Fig. 3.

The same NN configuration was used (one hidden layer and seven nodes) which was trained again with kp2 background events only but new setup additional setup cuts were applied (below is a list of all setup cuts, additional ones are the two last):

- fiducial ones
  - on Z_K
  - $P_K > 400$ GeV
- David's ones targeted odd and even high weight events
- $|M_{\gamma\gamma} - M_{\pi^0}| < 40$ MeV
- Lower-left part of the contour cut

The results with NN are shown in Fig. 4.

Unfortunately, I didn't cut off enough kcp3, ke3g with the additional setup cuts. Positive side is that kcp3 dropped an order of magnitude without degradation of the NN performance in respect to kp2.

Fig. 5 shows the results with harder setup cuts but without the NN retraining which would be appropriate.

Results with retrained NN are on the way...

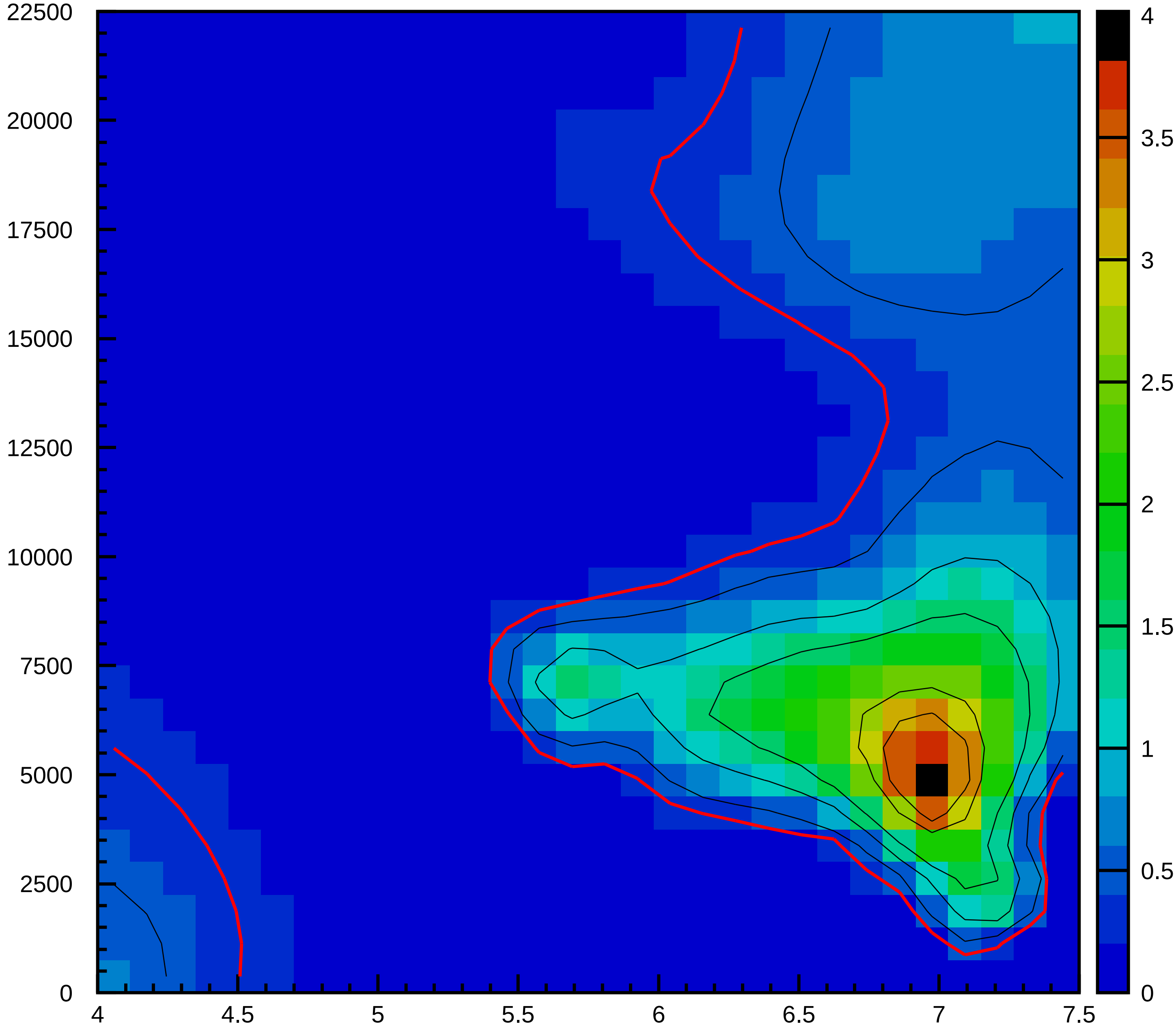

Figure 1: S/N ratio with T^2 vs $\log E_{miss}$ variables for tree major backgrounds (kp2, kcp3, ke3g)

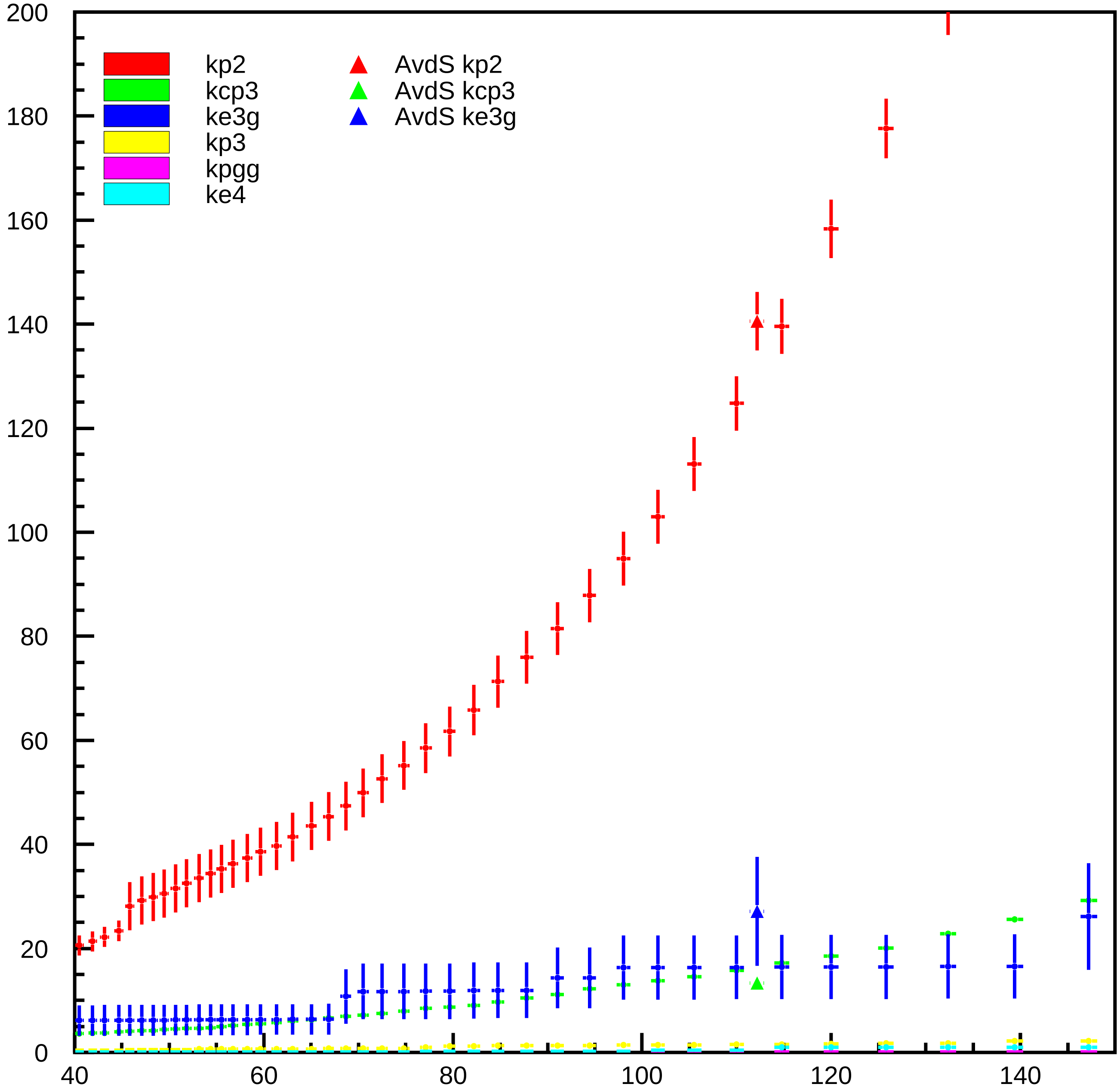

Figure 2: Normalized ($N_K = 0.1271 \times 10^{16}$ after the spoiler) background (see color legend) vs normalized signal. Triangular markers shows the results with Andrei's original contour.

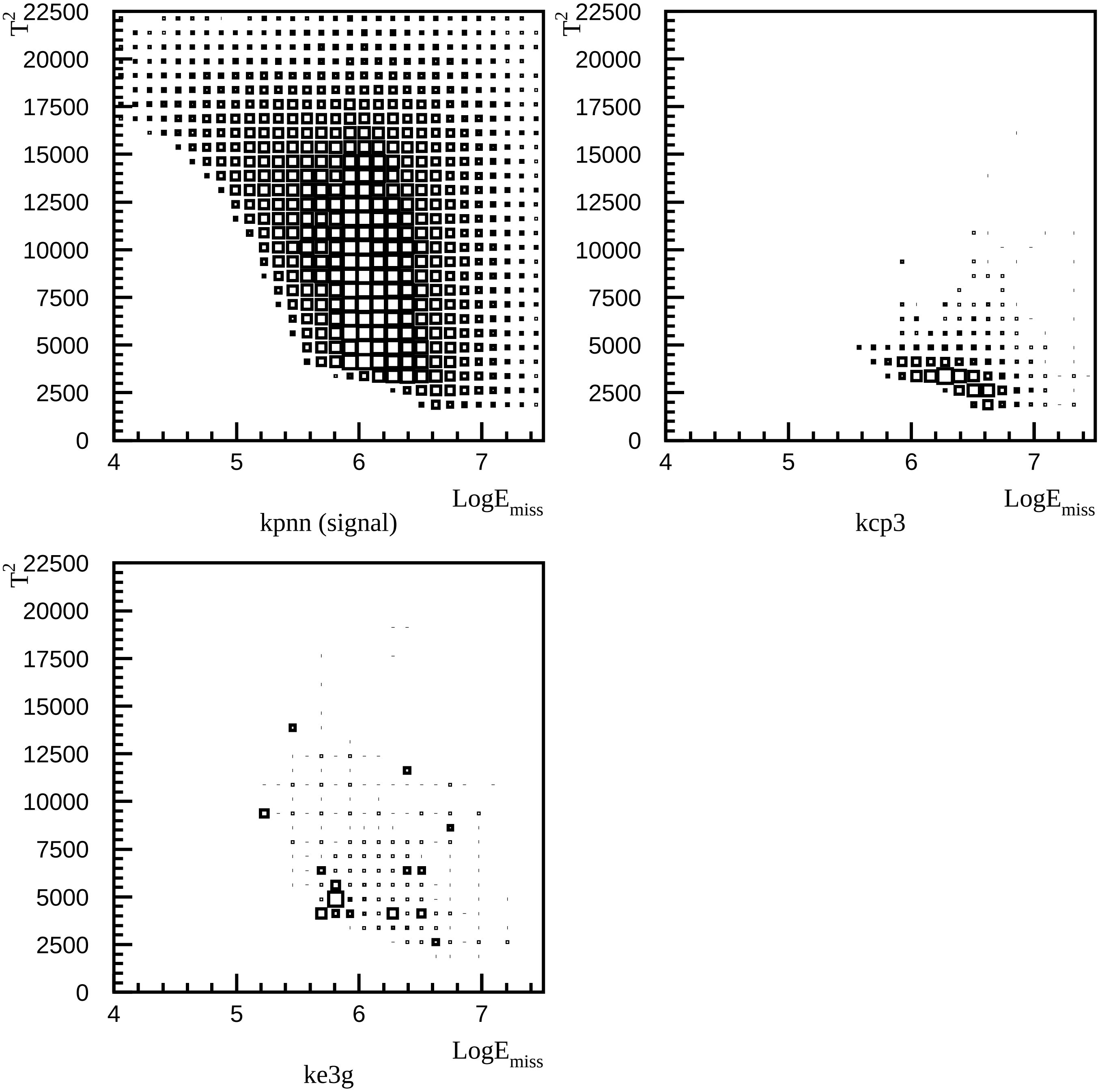

Figure 3: An additional NN setup cut: lower-left region on plots was cut off. Plots correspond to kpnn, kcp3 and ke3g events correspondingly.

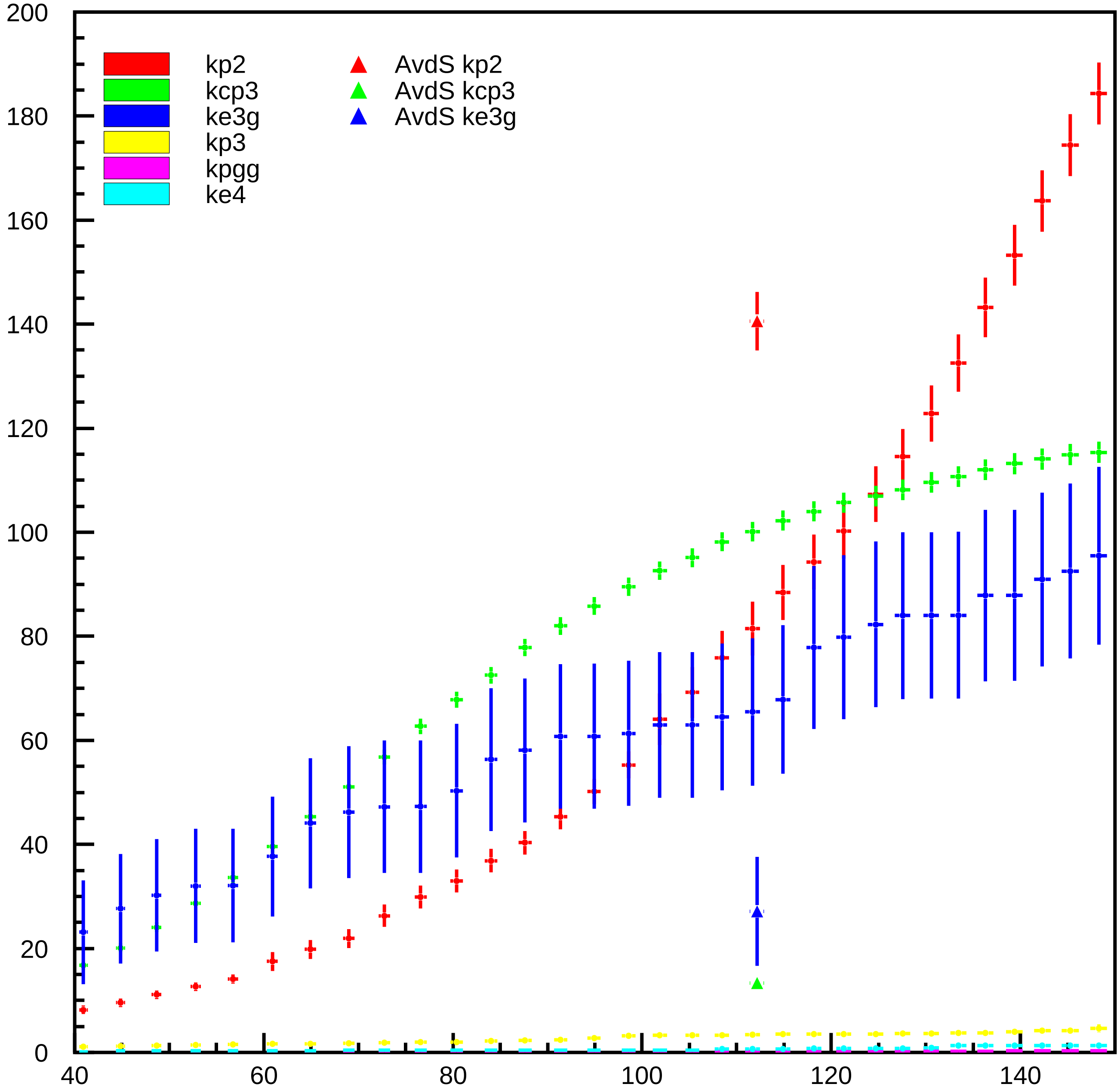

Figure 4: Normalized ($N_K = 0.1271 \times 10^{16}$ after the spoiler) background (see color legend) vs normalized signal obtained with NN.

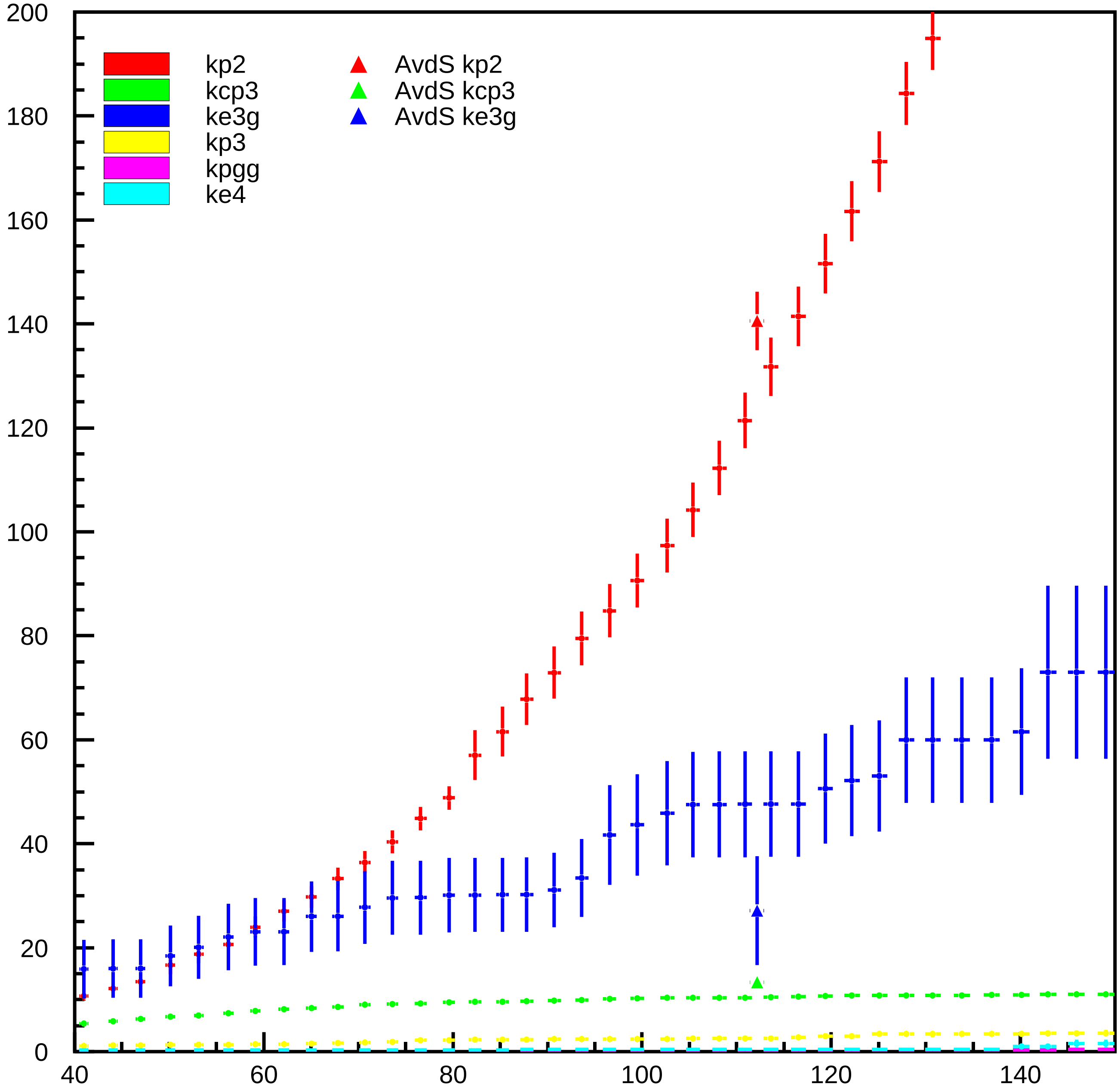

Figure 5: Normalized ($N_K = 0.1271 \times 10^{16}$ after the spoiler) background (see color legend) vs normalized signal obtained with NN. Harder setup cuts were applied but no NN retraining.